

Oracle® Fusion Middleware

Installing and Configuring Oracle GoldenGate for Sybase

12c (12.2.0.1)

E66359-03

February 2017

Contains system requirements, installation, and setup instructions for Oracle GoldenGate for the Sybase database.

Oracle Fusion Middleware Installing and Configuring Oracle GoldenGate for Sybase, 12c (12.2.0.1)

E66359-03

Copyright © 2013, 2017, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Related Documents	v
Conventions	vi
1 System Requirements and Preinstallation Instructions	
1.1 Verifying Certification and System Requirements	1-1
1.2 Operating System Requirements	1-1
1.2.1 Memory Requirements	1-1
1.2.2 Disk Requirements	1-2
1.2.3 Temporary Disk Requirements	1-2
1.2.4 Network Requirements	1-3
1.2.5 Operating System Privileges	1-3
1.2.6 Console Character Set	1-3
1.2.7 Other Programs	1-4
1.3 Database Requirements	1-4
1.3.1 Database Configuration	1-4
1.3.2 Database User for Oracle GoldenGate Processes	1-4
1.4 Supported Sybase Data Types	1-5
1.4.1 Integers	1-5
1.4.2 Floating-Point Numbers	1-6
1.4.3 Character Data	1-6
1.4.4 Dates and Timestamps	1-6
1.4.5 Large Objects	1-7
1.4.6 Money Types	1-8
1.4.7 IDENTITY Type	1-8
1.4.8 User-Defined Types	1-8
1.5 Non-Supported Sybase Data Types	1-8
1.6 Supported Operations and Objects for Sybase	1-8
1.7 Non-Supported Operations and Objects for Sybase	1-10
2 Installing Oracle GoldenGate	
2.1 Overview	2-1
2.2 Installation Overview	2-1

2.3	Understanding and Obtaining the Oracle GoldenGate Distribution	2-2
2.4	Setting Library Paths for Dynamic Builds on UNIX.....	2-2
2.5	Installing Oracle GoldenGate on Linux and UNIX.....	2-3
2.6	Installing Oracle GoldenGate on Windows	2-3
2.6.1	Installing the Oracle GoldenGate Files	2-4
2.6.2	Specifying a Custom Manager Name	2-4
2.6.3	Installing Manager as a Windows Service	2-4

3 Preparing the Sybase System for Oracle GoldenGate

3.1	Preparing Tables for Processing	3-1
3.1.1	Disabling Triggers and Cascade Constraints.....	3-1
3.1.2	Assigning Row Identifiers	3-2
3.1.2.1	How Oracle GoldenGate Determines the Kind of Row Identifier to Use	3-2
3.1.2.2	Using KEYCOLS to Specify a Custom Key.....	3-2
3.1.3	Limiting Row Changes in Tables that do not Have a Key.....	3-3
3.1.4	Replicating Encrypted Data	3-3
3.2	Preparing the Transaction Logs	3-3
3.2.1	Enabling Transaction Logging.....	3-3
3.2.2	Sizing and Retaining the Logs	3-4
3.2.3	Initializing the Secondary Truncation Point.....	3-4

4 Uninstalling Oracle GoldenGate

4.1	Uninstalling Oracle GoldenGate from Linux or UNIX	4-1
4.2	Uninstalling Oracle GoldenGate from Windows.....	4-2

A Oracle GoldenGate Installed Components

A.1	Oracle GoldenGate Programs And Utilities	A-1
A.2	Oracle GoldenGate Subdirectories	A-2
A.3	Other Oracle GoldenGate Files	A-4
A.4	Oracle GoldenGate Checkpoint Table	A-7

Preface

With Oracle GoldenGate for Sybase database, you can replicate data to and from supported Sybase versions or between a Sybase database and a database of another type. Oracle GoldenGate for Sybase supports data filtering, mapping, and transformation unless noted otherwise in this documentation.

This guide helps you get started with installing Oracle GoldenGate on a Sybase database system and performing initial setup. Refer to the other Oracle GoldenGate documentation listed in this Preface for additional information to configure, run, and manage your Oracle GoldenGate environment.

Audience

This guide is intended for installers, database administrators, and system administrators who are installing, configuring and running Oracle GoldenGate.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

The Oracle GoldenGate documentation set includes the following components:

Windows, UNIX, and Linux Platforms

- *Installing and Configuring Oracle GoldenGate for DB2 for i*
- *Installing and Configuring Oracle GoldenGate for DB2 LUW*
- *Installing and Configuring Oracle GoldenGate for DB2 z/OS*
- *Installing and Configuring Oracle GoldenGate for Informix*
- *Installing and Configuring Oracle GoldenGate for MySQL*

- *Installing and Configuring Oracle GoldenGate for NonStop SQL/MX*
- *Installing and Configuring Oracle GoldenGate for SQL Server*
- *Installing and Configuring Oracle GoldenGate for Oracle TimesTen*
- *Installing and Configuring Oracle GoldenGate for Oracle Database*
- *Installing and Configuring Oracle GoldenGate for Sybase*
- *Installing and Configuring Oracle GoldenGate for Teradata*
- *Administering Oracle GoldenGate for Windows and UNIX*
- *Reference for Oracle GoldenGate for Windows and UNIX*
- *Logdump Reference for Oracle GoldenGate*
- *Upgrading Oracle GoldenGate for Windows and UNIX*
- *Error Messages Reference for Oracle GoldenGate for Windows and UNIX*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select Save ." Boldface also is used for terms defined in text or in the glossary.
<i>italic</i> <i>italic</i>	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: <code>TABLE <i>table_name</i></code> . Italic type also is used for book titles and emphasis.
monospace MONOSPACE	Monospace type indicates code components such as user exits and scripts; the names of files and database objects; URL paths; and input and output text that appears on the screen. Uppercase monospace type is generally used to represent the names of Oracle GoldenGate parameters, commands, and user-configurable functions, as well as SQL commands and keywords.
UPPERCASE	Uppercase in the regular text font indicates the name of a utility unless the name is intended to be a specific case.
{ }	Braces within syntax enclose a set of options that are separated by pipe symbols, one of which must be selected, for example: <code>{<i>option1</i> <i>option2</i> <i>option3</i>}</code> .
[]	Brackets within syntax indicate an optional element. For example in this syntax, the <code>SAVE</code> clause is optional: <code>CLEANUP REPLICAT <i>group_name</i> [, <i>SAVE count</i>]</code> . Multiple options within an optional element are separated by a pipe symbol, for example: <code>[<i>option1</i> <i>option2</i>]</code> .

System Requirements and Preinstallation Instructions

This chapter contains the requirements for the system and database resources that support Oracle GoldenGate.

This chapter contains the following sections:

- [Section 1.1, "Verifying Certification and System Requirements"](#)
- [Section 1.2, "Operating System Requirements"](#)
- [Section 1.3, "Database Requirements"](#)
- [Section 1.4, "Supported Sybase Data Types"](#)
- [Section 1.5, "Non-Supported Sybase Data Types"](#)
- [Section 1.6, "Supported Operations and Objects for Sybase"](#)
- [Section 1.7, "Non-Supported Operations and Objects for Sybase"](#)

1.1 Verifying Certification and System Requirements

Make sure that you are installing your product on a supported hardware or software configuration. For more information, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

Oracle has tested and verified the performance of your product on all certified systems and environments; whenever new certifications occur, they are added to the proper certification document right away. New certifications can occur at any time, and for this reason the certification documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

1.2 Operating System Requirements

This section outlines the operating system resources that are necessary to support Oracle GoldenGate.

1.2.1 Memory Requirements

The amount of memory that is required for Oracle GoldenGate depends on the amount of data being processed, the number of Oracle GoldenGate processes running, the amount of RAM available to Oracle GoldenGate, and the amount of disk space that is available to Oracle GoldenGate for storing pages of RAM temporarily on disk when the operating system needs to free up RAM (typically when a low watermark is reached). This temporary storage of RAM to disk is commonly known as *swapping* or

paging (herein referred to as *swapping*). Depending on the platform, the term *swap space* can be a swap partition, a swap file, a page file (Windows) or a shared memory segment (IBM i platforms).

Modern servers have sufficient RAM combined with sufficient swap space and memory management systems to run Oracle GoldenGate. However, increasing the amount of RAM available to Oracle GoldenGate may significantly improve its performance, as well as that of the system in general.

Typical Oracle GoldenGate installations provide RAM in multiples of gigabytes to prevent excessive swapping of RAM pages to disk. The more contention there is for RAM the more swap space that is used.

Excessive swapping to disk causes performance issues for the Extract process in particular, because it must store data from each open transaction until a commit record is received. If Oracle GoldenGate runs on the same system as the database, the amount of RAM that is available becomes critical to the performance of both.

RAM and swap usage are controlled by the operating system, not the Oracle GoldenGate processes. The Oracle GoldenGate cache manager takes advantage of the memory management functions of the operating system to ensure that the Oracle GoldenGate processes work in a sustained and efficient manner. In most cases, users need not change the default Oracle GoldenGate memory management configuration.

For more information about evaluating Oracle GoldenGate memory requirements, see the `CACHEMGR` parameter in *Reference for Oracle GoldenGate for Windows and UNIX*.

1.2.2 Disk Requirements

Assign the following free disk space:

- To determine the size of the Oracle GoldenGate download file, view the Size column before downloading your selected build from Oracle Software Delivery Cloud. The value shown is the size of the files in compressed form. The size of the expanded Oracle GoldenGate installation directory will be significantly larger on disk. For more information, see [Section 2.3, "Understanding and Obtaining the Oracle GoldenGate Distribution."](#)
- Allow at least an additional 1 GB of disk space on any system that hosts Oracle GoldenGate trails, which are files that contain the working data. You may need more or less than this amount, because the space that is consumed by the trails depends on the volume of data that will be processed. See the guidelines for sizing trails in *Administering Oracle GoldenGate for Windows and UNIX*.

1.2.3 Temporary Disk Requirements

By default, Oracle GoldenGate maintains data that it swaps to disk in the `dirtmp` sub-directory of the Oracle GoldenGate installation directory. The cache manager assumes that all of the free space on the file system is available. This directory can fill up quickly if there is a large transaction volume with large transaction sizes. To prevent I/O contention and possible disk-related Extract failures, dedicate a disk to this directory. You can assign a name and size to this directory with the `CACHEDIRECTORY` option of the `CACHEMGR` parameter. The `CACHESIZE` option of `CACHEMGR` sets a soft limit for the amount of virtual memory (cache size) that is available for caching transaction data. See *Reference for Oracle GoldenGate for Windows and UNIX* for the default values of these options and detailed explanations, in case system adjustments need to be made.

1.2.4 Network Requirements

The following network resources must be available to support Oracle GoldenGate.

- Configure the system to use TCP/IP services, including DNS. Oracle GoldenGate supports IPv4 and IPv6 and can operate in a system that supports one or both of these protocols.
- Configure the network with the host names or IP addresses of all systems that will be hosting Oracle GoldenGate processes and to which Oracle GoldenGate will be connecting. Host names are easier to use.
- Oracle GoldenGate requires some unreserved and unrestricted TCP/IP ports, the number of which depends on the number and types of processes in your configuration. For details on how to configure the Manager process to handle the required ports, see *Administering Oracle GoldenGate for Windows and UNIX*.
- Keep a record of the ports that you assigned to Oracle GoldenGate. You will specify them with parameters when configuring the Manager process.
- Configure your firewalls to accept connections through the Oracle GoldenGate ports.

1.2.5 Operating System Privileges

The following are the privileges in the operating system that are required to install Oracle GoldenGate and to run the processes.

- To install on Windows, the person who installs Oracle GoldenGate must log in as Administrator.
- To install on UNIX, the person who installs Oracle GoldenGate must have read and write privileges on the Oracle GoldenGate installation directory.
- The Oracle GoldenGate Extract, Replicat, and Manager processes must operate as an operating system user that has privileges to read, write, and delete files and subdirectories in the Oracle GoldenGate directory. In addition, the Manager process requires privileges to control the other Oracle GoldenGate processes.
- The Extract process must operate as an operating system user that has read access to the transaction log files, both online and archived. If you install the Manager process as a Windows service during the installation steps in this documentation, you must install as Administrator for the correct permissions to be assigned. If you cannot install Manager as a service, assign read access to the Extract process manually, and then always run Manager and Extract as Administrator.
- Dedicate the Extract, Replicat, and Manager operating system users to Oracle GoldenGate. Sensitive information might be available to anyone who runs an Oracle GoldenGate process.

1.2.6 Console Character Set

The operating system and the command console must have the same character sets. Mismatches occur on Microsoft Windows systems, where the operating system is set to one character set, but the DOS command prompt uses a different, older DOS character set. Oracle GoldenGate uses the character set of the operating system to send information to GGSCI command output; therefore a non-matching console character set causes characters not to display correctly. You can set the character set of the console before opening a GGSCI session by using the following DOS command:

```
chcp OS_character_set
```

If the characters do not display correctly after setting the code page, try changing the console font to Lucida Console, which has an extended character set.

1.2.7 Other Programs

The following are additional considerations in support of Oracle GoldenGate.

- Before installing Oracle GoldenGate on a Windows system, install and configure the Microsoft Visual C++ 2010 SP1 Redistributable Package. Make certain it is the SP1 version of this package, and make certain to get the correct bit version for your server. This package installs runtime components of Visual C++ Libraries. For more information, and to download this package, go to <http://www.microsoft.com>.
- Oracle GoldenGate fully supports virtual machine environments created with any virtualization software on any platform. When installing Oracle GoldenGate into a virtual machine environment, select a build that matches the database and the operating system of the virtual machine, not the host system.

1.3 Database Requirements

This section contains Oracle GoldenGate requirements that are specific to the Sybase database.

1.3.1 Database Configuration

The Extract process makes calls directly to the Sybase Replication API on a source Sybase server. The source database on this server must be configured as follows to support data capture by Oracle GoldenGate.

- Set the `DSQUERY` variable to the server that contains the database that Oracle GoldenGate will be using.
- Because Extract uses the Sybase LTM to read the Sybase transaction log, it cannot run if Sybase RepServer is running on the same database. Only one process at a time can reserve a context that allows it to read the transaction log on the same database.
- The Extract process must be permitted to manage the secondary log truncation point. For more information, see [Section 3.2.3, "Initializing the Secondary Truncation Point"](#).
- Multi-site availability leverages SAP Sybase Rep Server Warm Standby, however, it is a secondary or tertiary or high standby. Oracle GoldenGate for Sybase cannot capture from the primary Warm Standby, but you can capture from the secondary (or higher standby) because SAP Sybase Rep Server is not in control of the Transaction Log for those database.
- Configure the database page size to 4k, 8k, 16k, 32k, or larger. The "upgradecheckpoint" table has row size of more than 2K pages so if it is programmatically created it will fail to do so in the 2K database page size.

1.3.2 Database User for Oracle GoldenGate Processes

Oracle GoldenGate requires a database user account. Create this account and assign privileges according to the following guidelines.

- To preserve the security of your data, and to monitor Oracle GoldenGate processing accurately, do not permit other users, applications, or processes to log on, or operate as, the Oracle GoldenGate database user.
- Create a database user that is dedicated to Oracle GoldenGate. It can be the same user for all of the Oracle GoldenGate processes that must connect to a database:
 - Extract (source database)
 - Replicat (target database)
 - DEFGEN utility (source or target database)
- The Extract process requires permission to access the source database. Do one of the following:
 - Grant System Administrator privileges.
 - Assign a user name with `replication_role`. The command to grant replication role is either:

Or

```
sp_role 'grant', replication_role, Extract_user
```

```
use dbname grant role replication_role to Extract_user
```

Note: Specific DDL or DML operations may require the use of both `sa_role` and `replication_role`.

- The Replicat process requires connect and DML privileges on the target database.

1.4 Supported Sybase Data Types

This section lists the Sybase data types that Oracle GoldenGate supports and any limitations of this support.

1.4.1 Integers

- BIGINT
- BIT
- DECIMAL
- INT (signed and unsigned)
- TINYINT (signed and unsigned)
- NUMERIC
- SMALLINT (signed and unsigned)

Limitations of Support

- NUMERIC and DECIMAL (fixed-point) are supported with no integrity loss when moving data to a target column of the same data type without involving calculations or transformation. When calculations or transformation must be performed, Oracle GoldenGate supports a maximum value of a signed long integer (32-bits).

- BIT is supported for automatic mapping between Sybase databases. To move BIT data between Sybase and another database type, Oracle GoldenGate treats BIT data as binary. In this case, the following are required:
 - The BIT column must be mapped to the corresponding source or target column with a COLMAP clause in a TABLE or MAP statement.
- For the Sybase 157 GA release, these data types cannot be replicated:
 - BIGINT (as a key column)
 - BIGDATETIME
 - BIGTIME
- When replicating TINYINT and Extract is not in the same version of Replicat, you will need to create a `sourcedef` and/or `targetdef` file even if you are replicating between identical Sybase versions.
- See also [Section 1.5, "Non-Supported Sybase Data Types"](#).

1.4.2 Floating-Point Numbers

- DOUBLE
- FLOAT
- REAL

Limitations of Support

The support of range and precision for floating-point numbers depends on the host machine. In general, the precision is accurate to 16 significant digits, but you should review the database documentation to determine the expected approximations. Oracle GoldenGate rounds or truncates values that exceed the supported precision.

1.4.3 Character Data

- CHAR
- NCHAR
- NVARCHAR
- VARCHAR
- UNICHAR
- UNIVARCHAR

Limitations of Support

- These data types are supported to the maximum length supported by the database, this being the maximum page size.
- Fetching NVARCHAR replication results using the Sybase `char_length` or `datalength` functions when a Sybase database is the target and the source is a heterogeneous database and you replicate from the source to the target may result in a data integrity issue. This occurs when you use a Sybase release earlier than Adaptive Server Enterprise 15.5 for Windows x64 platform EBF 21262: 15.5 ESD #5.3.

1.4.4 Dates and Timestamps

- BIGDATETIME

- BIGTIME
- DATE
- DATETIME
- SMALLDATETIME
- TIME

Limitations of Support

- Oracle GoldenGate supports timestamp data from 0001/01/03:00:00:00 to 9999/12/31:23:59:59. If a timestamp is converted from GMT to local time, these limits also apply to the resulting timestamp. Depending on the time zone, conversion may add or subtract hours, which can cause the timestamp to exceed the lower or upper supported limit.
- Oracle GoldenGate does not support negative dates.

1.4.5 Large Objects

- BINARY
- IMAGE
- TEXT
- UNITEXT
- VARBINARY

Limitations of Support

- TEXT, UNITEXT and IMAGE are supported up to 2 GB in length.
- Large objects that are replicated from other databases (such as Oracle BLOB and CLOB) can be mapped to Sybase CHAR, VARCHAR, BINARY, and VARBINARY columns. To prevent Replicat from abending if the replicated large object is bigger than the size of the target column, use the DBOPTIONS parameter with the ALLOWLOBDATATRUNCATE option in the Replicat parameter file. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.
- To move data to a Sybase target from a source database that permits empty LOB columns, use the DBOPTIONS parameter with the EMPTYLOBSTRING option in the Replicat parameter file. This parameter accepts a string value and prevents Replicat from setting the target column to NULL, which is not permitted by Sybase. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.
- When a source table contains multiple identical rows, it can cause LOB inconsistencies in the target table. This occurs when the source table lacks a primary key or other unique row identifier. The rows are inserted by Replicat on the target, but if the LOB data is updated in a subsequent source operation, it will only be replicated to the first row that was inserted on the target.
- Do not use NOT NULL constraints on the in-row LOB column. If you want to use NOT NULL constraints, use them on the off-row LOB column.
- If you need to fetch the in-row LOB data directly from the table you must use FETCHCOLS/FETCHMODCOLS.
- Oracle GoldenGate for Sybase 15.7 does not support the in-row LOB column replication (however, it can still push the data into the in-row LOB column at in the Replicat database). This means tables included in the replication cannot have

any in-row LOB columns. Oracle GoldenGate will abend if any replication table includes an in-row LOB column. If you need in-row LOB support, contact Oracle Support for further information.

1.4.6 Money Types

- MONEY
- SMALLMONEY

Limitations of Support

Money data types are supported with no integrity loss when moving data to a target column of the same data type without involving calculations or transformation. When calculations or transformation must be performed, Oracle GoldenGate supports a maximum value of a signed long integer (32-bits).

1.4.7 IDENTITY Type

The `IDENTITY` data type is supported for replication in one direction only, but not for a bi-directional configuration.

1.4.8 User-Defined Types

User-defined types are fully supported.

1.5 Non-Supported Sybase Data Types

This section lists the Sybase data types that Oracle GoldenGate does not support.

- The `TIMESTAMP` data is not supported. Timestamp columns data is captured though the data cannot be applied to the Sybase timestamp column due to a database limitation. The database populates this column automatically once that corresponding row is inserted or updated. To exclude timestamp columns from being captured by Oracle GoldenGate, use the `COLSEXCEPT` option of the `TABLE` parameter. Because the system generates the timestamps, the source and target values will be different.
- The Java `rowobject` data type is not supported.

1.6 Supported Operations and Objects for Sybase

This section lists the data operations and database objects that Oracle GoldenGate supports.

- Oracle GoldenGate supports the extraction and replication of insert, update, and delete operations on Sybase tables that contain rows of up to 512 KB in length.
- Oracle GoldenGate supports the maximum number of columns and the maximum column size per table that is supported by the database.
- Oracle GoldenGate supports deferred inserts, deferred indirect inserts, deferred updates, and deferred deletes. It is possible that the use of deferred updates could cause primary key constraint violations for the affected SQL on the target. If these errors occur, use the `Replicat` parameter `HANDLECOLLISIONS`.
- Oracle GoldenGate supports `TRUNCATE TABLE` if the names of the affected tables are unique across all schemas. If the table names are not unique across all schemas,

use the `IGNORETRUNCATES` parameter for those tables to prevent Replicat from abending.

- Oracle GoldenGate supports `GETTRUNCATES` and `IGNORETRUNCATES` by Extract and Replicat.
- Oracle GoldenGate supports data that is encrypted with a system-encrypted password.
- Oracle GoldenGate supports array fetching during initial loads, as controlled by the `FETCHBATCHSIZE` parameter.
- The `BATCHSQL` Replicat feature of Oracle GoldenGate is supported on ASE 15.7 SP110 and greater on the following platforms:
 - AIX
 - Linux x64
 - Sun Solaris SPARC
 - Sun Solaris x64
 - Windows x64

The Sybase specific parameter `sybIgnoreConvError` is not supported with `BatchSQL` feature. In certain scenarios, the `CS_NUMERIC` and `CS_DECIMAL` data types are not supported by `BatchSQL` because of a bug in the Sybase specific CT Library. LOB replication is supported in `BatchSql` mode for Sybase database version 157 SP110 onward. This will improve the LOB replication performance. It is restricted to 16384 bytes of LOB data that means if LOB data is more than 16384 bytes, the data would not be processed through `BATCHSQL` mode instead the mode switched to Normal.

- Limitations on Computed Columns support are as follows:
 - Oracle GoldenGate fully supports persisted computed columns. The change values are present in the transaction log and can be captured to the trail.
 - You cannot use `NOT NULL` constraints on in-row LOB columns. If you need to use `NOT NULL` constraints, do so only with off-row LOB columns.
 - Oracle GoldenGate supports tables with non-persisted computed columns, but does not capture change data for these columns because the database does not write it to the transaction log. To replicate data for non-persisted computed columns, use the `FETCHCOLS` or `FETCHMODCOLS` option of the `TABLE` parameter to fetch the column data from the table. Keep in mind that there can be discrepancies caused by differences in data values between when the column was changed in the database and when Extract fetches the data for the transaction record that is being processed.
 - Replicat does not apply DML to any computed column, even if the data for that column is in the trail, because the database does not permit DML on that type of column. Data from a source *persisted* computed column, or from a fetched non-persisted column, can be applied to a target column that is not a computed column.
 - In an initial load, all of the data is selected directly from the source tables, not the transaction log. Therefore, in an initial load, data values for all columns, including non-persisted computed columns, gets written to the trail or sent to the target, depending on the method that is being used. As when applying change data, however, Replicat does not apply initial load data to computed columns, because the database does not permit DML on that type of column.

- Oracle GoldenGate will not use a persisted computed column that is defined as a key column, an index column, or that is part of a `KEYCOLS` clause in a `TABLE` or `MAP` statement. If a unique key or index includes a computed column and Oracle GoldenGate must use that key, the computed column will be ignored. Additionally, if a unique key or index contains a computed column and is the only unique identifier on the table, Oracle GoldenGate will use all of the columns except the computed column as an identifier to find the target row. Thus, the presence of a computed column in a key or index affects data integrity if the remaining columns do not enforce uniqueness. Note that Sybase does not support non-persisted computed columns as part of a key, and neither does Oracle GoldenGate.
- For Oracle GoldenGate to support `TRUNCATE TABLE`, all table names should be unique across all schemas within a database. This rule applies to Extract and Replicat.
- Limitations on Automatic Heartbeat Table support are as follows:
 - Heartbeat frequency should be an integer that is divisible by 60. Oracle GoldenGate heartbeat parameter frequency is accepted in minutes, although you can use in seconds. The Sybase job scheduler uses the minutes in integer not in decimal so it is converted internally to set the frequency in minutes to the nearest possible value. For example, setting this value to 65 seconds results in the frequency being set to 1 minute; 140 seconds results in the value set to 2 minutes.
 - Data truncation occurs with a Replicat abend when it exceeds more than 1500 characters for the `incoming_routing_path` and `outgoing_routing_path` of the `GG_HEARTBEAT_SEED`, `GG_HEARTBEAT`, and `GG_HEARTBEAT_HISTORY` tables. The `incoming_routing_path` and `outgoing_routing_path` size of these tables is set to 1500 characters in ASCII and is a 500 max bytes in multibyte characters. Ensure that the incoming and outgoing routing path strings are within the specified limit.
 - The default Oracle GoldenGate schema, `GGUSER`, is not supported. For Heartbeat Table functionality to operate correctly, the login user must have the `replication_role`, `js_admin_role`, `js_user_role` roles.

1.7 Non-Supported Operations and Objects for Sybase

This section lists the data operations and database objects that Oracle GoldenGate does not support.

- Data that is encrypted with a user-defined password.
- Extraction or replication of DDL (data definition language) operations.
- Multi-Extract configuration. Only one Extract can reserve a context to read the Sybase transaction logs.
- Because `SHOWSYNTAX` is supported in the `DYNSQL` mode, `NODYNSQL` is deprecated.
- Table names that contain data with an underscore followed by some characters then a space (for example, `'zzz_j '`) is not supported. Oracle GoldenGate cannot process records containing this type of character string with `GGSCI`, `DEFGEN`, `EXTRACT`, or `REPLICAT`. Additionally, this type of data cannot be used with Oracle GoldenGate wildcard (*). If you do have this type of data in your table name, you must drop this kind of table name from your database, and then they restart the application to process and respect Oracle GoldenGate wildcard.

Installing Oracle GoldenGate

This chapter contains instructions for installing Oracle GoldenGate for Sybase for the first time. It contains the following sections:

- [Section 2.1, "Overview"](#)
- [Section 2.2, "Installation Overview"](#)
- [Section 2.3, "Understanding and Obtaining the Oracle GoldenGate Distribution"](#)
- [Section 2.4, "Setting Library Paths for Dynamic Builds on UNIX"](#)
- [Section 2.5, "Installing Oracle GoldenGate on Linux and UNIX"](#)
- [Section 2.6, "Installing Oracle GoldenGate on Windows"](#)

These instructions are for installing Oracle GoldenGate for the first time. Additionally, they are for downloading the base release of a new version of Oracle GoldenGate.

To download and install subsequent patches to the base release, go to the Patches and Updates tab of My Oracle Support at:

<http://support.oracle.com>

To upgrade Oracle GoldenGate from one version to another, follow the upgrade instructions at:

<http://docs.oracle.com/goldengate/c1221/gg-winux/index.html>

2.1 Overview

Installing Oracle GoldenGate installs all of the components that are required to run and manage the processing (excluding any components required from other vendors, such as drivers or libraries) and it installs the Oracle GoldenGate utilities.

2.2 Installation Overview

To install Oracle GoldenGate, the following steps are required:

- [Understanding and Obtaining the Oracle GoldenGate Distribution](#)
- [Setting Library Paths for Dynamic Builds on UNIX](#)
- [Installing Oracle GoldenGate on Linux and UNIX](#)
- [Installing Oracle GoldenGate on Windows](#)

2.3 Understanding and Obtaining the Oracle GoldenGate Distribution

For complete information about how to obtain Oracle Fusion Middleware software, see "Understanding and Obtaining Product Distributions" in *Planning an Installation of Oracle Fusion Middleware*.

To download the Oracle WebLogic Server and Coherence software for development or evaluation, see the following location on the Oracle Technology Network (OTN):

<http://www.oracle.com/technetwork/middleware/fusion-middleware/downloads/index.html>

For more information about locating and downloading Oracle Fusion Middleware products, see the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on OTN.

ORIGINAL TEXT BELOW; keep file name???

To obtain Oracle GoldenGate follow these steps:

1. Go to Oracle Technology Network.
2. Find the Oracle GoldenGate 12c (12.2.0.1) release and download the ZIP file onto your system.

2.4 Setting Library Paths for Dynamic Builds on UNIX

Oracle GoldenGate uses shared libraries. When you install Oracle GoldenGate on a UNIX system, the following must be done *before* you run GGSCI or any other Oracle GoldenGate process . If you will be running an Oracle GoldenGate program from outside the Oracle GoldenGate installation directory on a UNIX system:

- (Optional) Add the Oracle GoldenGate installation directory to the `PATH` environment variable.
- (Required) Add the Oracle GoldenGate installation directory to the `shared-libraries` environment variable.

For example, given an Oracle GoldenGate installation directory of `/users/ogg`, the second command in the following example requires these variables to be set:

Table 2–1 Command Requiring Library Variable

Command	Requires GG libraries in environment variable?
<code>\$ users/ogg > ./ggsci</code>	No
<code>\$ users > ./ogg/ggsci</code>	Yes

To Set the Variables in Korn Shell:

```
PATH=installation_directory:$PATH
export PATH
shared_libraries_variable=absolute_path_of_installation_directory:$shared_libraries_variable
export shared_libraries_variable
```

To Set the Variables in Bourne Shell:

```
export PATH=installation_directory:$PATH
export shared_libraries_variable=absolute_path_of_installation_directory:$shared_libraries_variable
```

To Set the Variables in C Shell:

```
setenv PATH installation_directory:$PATH
setenv shared_libraries_variable absolute_path_of_installation_directory:$shared_libraries_variable
```

Where: *shared_libraries_variable* is one of the variables shown in [Table 2-2](#):

Table 2-2 UNIX/Linux Library Path Variables per Platform

Platform ¹	Environment variable
IBM AIX	LIBPATH
IBM z/OS	
HP-UX	SHLIB_PATH
Sun Solaris	LD_LIBRARY_PATH ²
LINUX	

¹ A specific platform may or may not be supported by Oracle GoldenGate for your database.

² In 64-bit environments with 32-bit Oracle databases, Oracle GoldenGate requires the LD_LIBRARY_PATH to include the 32-bit Oracle libraries.

Example

```
export LD_LIBRARY_PATH=/ggs/12.0:$LD_LIBRARY_PATH
```

Note: To view the libraries that are required by an Oracle GoldenGate process, use the `ldd goldengate_process` shell command before starting the process. This command also shows an error message for any that are missing.

2.5 Installing Oracle GoldenGate on Linux and UNIX

Follow these steps to install Oracle GoldenGate for Oracle on a Linux or UNIX system.

1. Extract the Oracle GoldenGate `mediapack.zip` file to the system and directory where you want Oracle GoldenGate to be installed.
2. Run the command shell.
3. Change directories to the new Oracle GoldenGate directory.
4. From the Oracle GoldenGate directory, run the GGSCI program.

```
GGSCI
```

5. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

```
CREATE SUBDIRS
```

6. Issue the following command to exit GGSCI.

```
EXIT
```

2.6 Installing Oracle GoldenGate on Windows

Follow these steps to install Oracle GoldenGate for Oracle on a Windows system.

[Section 2.6.1, "Installing the Oracle GoldenGate Files"](#)

[Section 2.6.2, "Specifying a Custom Manager Name"](#)

[Section 2.6.3, "Installing Manager as a Windows Service"](#)

2.6.1 Installing the Oracle GoldenGate Files

1. Unzip the downloaded file(s) by using WinZip or an equivalent compression product.
2. Move the files in binary mode to a folder on the drive where you want to install Oracle GoldenGate. *Do not* install Oracle GoldenGate into a folder that contains spaces in its name, even if the path is in quotes. For example:

```
C:\ "Oracle GoldenGate" is not valid.
```

```
C:\Oracle_GoldenGate is valid.
```

3. From the Oracle GoldenGate folder, run the GGSCI program.
4. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

```
CREATE SUBDIRS
```

5. Issue the following command to exit GGSCI.

```
EXIT
```

2.6.2 Specifying a Custom Manager Name

You must specify a custom name for the Manager process if either of the following is true:

- You want to use a name for Manager other than the default of GGSMGR .
- There will be multiple Manager processes running as Windows services on this system. Each Manager on a system must have a unique name. Before proceeding further, note the names of any local Manager services.

1. From the directory that contains the Manager program, run GGSCI.
2. Issue the following command.

```
EDIT PARAMS ./GLOBALS
```

Note: The `./` portion of this command must be used, because the `GLOBALS` file must reside at the root of the Oracle GoldenGate installation file.

3. In the file, add the following line, where *name* is a one-word name for the Manager service.

```
MGRSERVNAME name
```

4. Save the file. The file is saved automatically with the name `GLOBALS`, but without a file extension. Do not move this file. It is used during installation of the Windows service and during data processing.

2.6.3 Installing Manager as a Windows Service

By default, Manager is not installed as a service and can be run by a local or domain account. However, when run this way, Manager will stop when the user logs out.

When you install Manager as a service, you can operate it independently of user connections, and you can configure it to start manually or at system start-up.

1. (Recommended) Log on as the system administrator.
2. Click **Start** then **Run** and type `cmd` in the Run dialog box.
3. From the directory that contains the Manager program that you are installing as a service, run the `install` utility with the following syntax:

```
install option [...]
```

Where: *option* is one of the following:

Table 2–3 *install Options*

Option	Description
ADDEVENTS	Adds Oracle GoldenGate events to the Windows Event Manager.
ADDSERVICE	Adds Manager as a service with the name that is specified with the <code>MGRSERVNAME</code> parameter in the <code>GLOBALS</code> file, if one exists, or by the default of <code>GGSMGR</code> . <code>ADDSERVICE</code> configures the service to run as the Local System account, the standard for most Windows applications because the service can be run independently of user logins and password changes. To run Manager as a specific account, use the <code>USER</code> and <code>PASSWORD</code> options. (Note that a user account can be changed by selecting the Properties action from the Services applet of the Windows Control Panel.) The service is installed to start at system boot time (see <code>AUTOSTART</code>). To start it after installation, either reboot the system or start the service manually from the Services applet of the Control Panel.
AUTOSTART	Sets the service that is created with <code>ADDSERVICE</code> to start at system boot time. This is the default unless <code>MANUALSTART</code> is used.
MANUALSTART	Sets the service that is created with <code>ADDSERVICE</code> to start manually through <code>GGSCI</code> , a script, or the Services applet of the Control Panel. The default is <code>AUTOSTART</code> .
USER <i>name</i>	Specifies a domain user account that executes Manager. For the <i>name</i> , include the domain name, a backward slash, and the user name, for example <code>HEADQT\GGSMGR</code> . By default, the Manager service is installed to use the Local System account.
PASSWORD <i>password</i>	Specifies the password for the user that is specified with <code>USER</code> .

4. If Windows User Account Control (UAC) is enabled, you are prompted to allow or deny the program access to the computer. Select **Allow** to enable the `install` utility to run.

The `install` utility installs the Manager service with a local system account running with administrator privileges. No further UAC prompts will be encountered when running Manager if installed as a service.

Note: If Manager is not installed as a service, Oracle GoldenGate users will receive a UAC prompt to confirm the elevation of privileges for Manager when it is started from the `GGSCI` command prompt. Running other Oracle GoldenGate programs also triggers a prompt.

Preparing the Sybase System for Oracle GoldenGate

This chapter contains guidelines for preparing the database and the system to support Oracle GoldenGate.

This chapter contains the following sections:

- [Section 3.1, "Preparing Tables for Processing"](#)
- [Section 3.2, "Preparing the Transaction Logs"](#)

3.1 Preparing Tables for Processing

The following table attributes must be addressed in an Oracle GoldenGate environment.

- [Disabling Triggers and Cascade Constraints](#)
- [Assigning Row Identifiers](#)
- [Limiting Row Changes in Tables that do not Have a Key](#)
- [Replicating Encrypted Data](#)

3.1.1 Disabling Triggers and Cascade Constraints

Disable triggers, cascade delete constraints, and cascade update constraints on target Sybase tables, or alter them to ignore changes made by the Oracle GoldenGate database user. Oracle GoldenGate replicates DML that results from a trigger or cascade constraint. If the same trigger or constraint gets activated on the target table, it becomes redundant because of the replicated version, and the database returns an error. Consider the following example, where the source tables are `emp_src` and `salary_src` and the target tables are `emp_targ` and `salary_targ`.

1. A delete is issued for `emp_src`.
2. It cascades a delete to `salary_src`.
3. Oracle GoldenGate sends both deletes to the target.
4. The parent delete arrives first and is applied to `emp_targ`.
5. The parent delete cascades a delete to `salary_targ`.
6. The cascaded delete from `salary_src` is applied to `salary_targ`.
7. The row cannot be located because it was already deleted in step 5.

To configure Replicat to disable target triggers at the start of its database session, take the following steps:

1. Assign the Replicat user the replication role.
2. Add the following parameter statement to the root level of the Replicat parameter file.

```
SQLEXEC "set triggers off"
```

3.1.2 Assigning Row Identifiers

Oracle GoldenGate requires some form of unique row identifier on source and target tables to locate the correct target rows for replicated updates and deletes.

3.1.2.1 How Oracle GoldenGate Determines the Kind of Row Identifier to Use

Unless a `KEYCOLS` clause is used in the `TABLE` or `MAP` statement, Oracle GoldenGate selects a row identifier to use in the following order of priority:

1. Primary key.
2. First unique key alphanumerically with no computed columns, no function-based columns, and no nullable columns.
3. First unique key alphanumerically with no computed or function-based columns, but can include nullable columns.
4. If none of the preceding key types exist (even though there might be other types of keys defined on the table) GoldenGate constructs a pseudo key of all columns that the database allows to be used in a unique key, excluding computed columns, function-based columns, and any columns that are explicitly excluded from the GoldenGate configuration.

Note: If there are other, non-usable keys on a table or if there are no keys at all on the table, Oracle GoldenGate logs an appropriate message to the report file. Constructing a key from all of the columns impedes the performance of Oracle GoldenGate on the source system. On the target, this key causes Replicat to use a larger, less efficient `WHERE` clause.

Oracle GoldenGate does not support a primary or unique key in which a column name contains an embedded comma, such as the following:

```
CREATE TABLE Tab ... PRIMARY KEY ("col1, Id", col2)
```

In this case "col1, ID" is the first primary key column and col2 is the second primary key column.

3.1.2.2 Using KEYCOLS to Specify a Custom Key

If a table does not have one of the preceding types of row identifiers, or if you prefer those identifiers not to be used, you can define a substitute key if the table has columns that always contain unique values. You define this substitute key by including a `KEYCOLS` clause within the Extract `TABLE` parameter and the Replicat `MAP` parameter. Oracle GoldenGate uses the specified key in place of any existing primary or unique key that it finds. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

3.1.3 Limiting Row Changes in Tables that do not Have a Key

If a target table has no primary key or unique key, duplicate rows can exist. It is possible for Oracle GoldenGate to update or delete too many rows in the target table, causing the source and target data to go out of synchronization without error messages to alert you. To limit the number of rows that are updated, use the `DBOPTIONS` parameter with the `LIMITROWS` option in the Replicat parameter file. `LIMITROWS` can increase the performance of Oracle GoldenGate on the target system because only one row is processed.

3.1.4 Replicating Encrypted Data

Oracle GoldenGate supports columns that are encrypted with a system-encrypted password, but not columns that are encrypted with a user-defined password. Check the tables from which you want to capture data against the following Oracle GoldenGate limitations:

- The table that contains the encrypted columns must have a primary or unique key.
- Columns that use encryption cannot be part of the primary key.

Encrypted columns are encrypted in the data files and in the log, so Extract must be configured to fetch the clear-text values from the database. To trigger this fetch, use the `FETCHCOLS` and `FETCHMODCOLS [EXCEPT]` options of the Extract `TABLE` parameter. `FETCHCOLS` forces a fetch of values that are not in the log, and `FETCHMODCOLS` or `FETCHMODCOLS [EXCEPT]` forces a fetch of values that are in the logs. Used together, these parameters ensure that the encrypted columns are always fetched from the database.

The following is an example of how to configure Extract to support the encryption. In this example, the encrypted column is `cardnum`.

```
TABLE ab.payments, FETCHCOLS (cardnum), FETCHMODCOLS (cardnum);
```

3.2 Preparing the Transaction Logs

To capture DML operations, Oracle GoldenGate reads the online logs. To ensure the continuity and integrity of Oracle GoldenGate processing, the logs must be configured as directed in the following sections:

[Section 3.2.1, "Enabling Transaction Logging"](#)

[Section 3.2.2, "Sizing and Retaining the Logs"](#)

[Section 3.2.3, "Initializing the Secondary Truncation Point"](#)

3.2.1 Enabling Transaction Logging

Use the `ADD TRANDATA` command to mark each source table for replication. This command uses the Sybase `sp_setreptable` and `sp_setrepcol` system procedures. `ADD TRANDATA` is the recommended way to mark the tables, instead of using those procedures through the database interface, but the owner or the system administrator can use them if needed. For more information, see the Sybase documentation.

To mark tables for replication with `ADD TRANDATA`:

1. On the source system, run `GGSCI` from the Oracle GoldenGate directory.
2. Log into the database from `GGSCI`.

```
DBLOGIN SOURCEDB database USERID user PASSWORD xxx
```

Where:

- *database* is the name of the database.
 - *user* is the database owner or the system administrator. You will be prompted for the password. This command has encryption options for the password. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.
 - *xxx* is the password for the associated *user*.
3. Issue the ADD TRANDATA command for each table to be marked.

```
ADD TRANDATA SCHEMA.TABLE LOBSNEVER | LOBSALWAYS | LOBSALWAYSNOINDEX |
LOBSIFCHANGED
```

Where:

- LOBSNEVER | LOBSALWAYS | LOBSALWAYSNOINDEX | LOBSIFCHANGED control whether LOB data is never propagated, only propagated if changed (the default), or always propagated. The ADD TRANDATA command will overwrite the LOB replication setting that is currently set for the table.

Note: Some ADD TRANDATA options enable the ALWAYS_REPLICATE option of sp_setrepcol. If a LOB column contains a NULL value, and then another column in the table gets updated (but not the LOB), that LOB will not be captured even though ALWAYS_REPLICATE is enabled.

3.2.2 Sizing and Retaining the Logs

Retain enough log data on the source system so that Extract can start again from its checkpoints after you stop it or there is an unplanned outage. Extract must have access to the log that contains the start of the oldest uncommitted unit of work, and all logs thereafter. To determine where the Extract checkpoints are, use the INFO EXTRACT command. For more information about INFO EXTRACT, see *Reference for Oracle GoldenGate for Windows and UNIX*.

If data that Extract needs during processing is not retained, either in online or backup logs, one of the following corrective actions might be required:

- You might need to alter Extract to capture from a later point in time for which log data is available (and accept possible data loss on the target).
- You might need to resynchronize the source and target tables, and then start the Oracle GoldenGate environment over again.

Make certain not to use backup or archive options that cause old archive files to be overwritten by new backups on the source system. New backups should be separate files with different names from older ones. This ensures that if Extract looks for a particular log, it will still exist, and it also ensures that the data is available in case it is needed for a support case.

3.2.3 Initializing the Secondary Truncation Point

Establish a secondary log truncation point on the source system prior to running the Oracle GoldenGate Extract process. Extract uses the secondary truncation point to identify data that remains to be processed.

To initialize the secondary truncation point, log on to the database as a user with `sa_` role privileges and then issue the following command:

```
dbcc settrunc( 'ltm', valid )
```

By default, Extract will manage the secondary truncation point once it is established. Do not permit Extract to be stopped any longer than necessary; otherwise the log could eventually fill up and the database will halt. The only way to resolve this problem is to disable the secondary truncation point and manage it outside of Oracle GoldenGate, and then purge the transaction log. Data not yet processed by Extract will be lost, and you will have to resynchronize the source and target data.

To control how the secondary truncation point is managed, use the `TRANLOGOPTIONS` parameter. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

Uninstalling Oracle GoldenGate

This chapter describes how to uninstall Oracle GoldenGate for Sybase. It contains the following sections

- [Section 4.1, "Uninstalling Oracle GoldenGate from Linux or UNIX"](#)
- [Section 4.2, "Uninstalling Oracle GoldenGate from Windows"](#)

This procedure assumes that you no longer need the data in the Oracle GoldenGate trails, and that you no longer need to preserve the current Oracle GoldenGate environment. To preserve your current environment and data, make a backup of the Oracle GoldenGate directory and all subdirectories before starting this procedure.

4.1 Uninstalling Oracle GoldenGate from Linux or UNIX

On all Systems:

1. Run the command shell.
2. (Suggested) Log on as the system administrator or as a user with permission to issue Oracle GoldenGate commands and delete files and directories from the operating system.
3. Change directories to the Oracle GoldenGate installation directory.
4. Run GGSCI.
5. Stop all Oracle GoldenGate processes.
6. Stop the Manager process.

On a Source System:

1. In GGSCI, log into the database with the `DBLOGIN` command.
2. In GGSCI, issue the `DELETE TRANDATA` command to disable replication.
3. If Extract was managing the secondary truncation point, enable the database to assume management of it.

```
dbcc settrunc( 'ltm', ignore )
```

On any System Where a Replicat Checkpoint Table Exists:

1. In GGSCI, log into the database with the `DBLOGIN` command.
2. In GGSCI, issue the `DELETE CHECKPOINTTABLE` command.

On all Systems:

1. Make certain that all processes are stopped, including GGSCI.
2. Remove the Oracle GoldenGate files by removing the installation directory.

4.2 Uninstalling Oracle GoldenGate from Windows

On all Systems:

1. (Suggested) Log on as the system administrator or as a user with permission to issue Oracle GoldenGate commands and to delete files and directories from the operating system.
2. From the Oracle GoldenGate installation folder, run GGSCI.
3. Stop all Oracle GoldenGate processes.
4. Stop the Manager program or service.
5. Run the Windows command console.
6. Change directories to the Oracle GoldenGate installation directory.
7. Run the `install` utility with the following syntax to stop the reporting of Oracle GoldenGate events to the Windows Event Manager and remove the Manager service.

```
install deleteevents deleteservice
```

On a Source System:

1. In GGSCI, log into the database with the `DBLOGIN` command.
2. In GGSCI, issue the `DELETE TRANDATA` command to disable replication.
3. If Extract was managing the secondary truncation point, enable the database to assume management of it.

```
dbcc settrunc( 'ltm', ignore )
```

On any System Where a Replicat Checkpoint Table Exists:

1. In GGSCI, log into the database with the `DBLOGIN` command.
2. In GGSCI, issue the `DELETE CHECKPOINTTABLE` command.

On all Systems:

1. Make certain that all processes are stopped, including GGSCI.
2. Remove the Oracle GoldenGate files by removing the installation directory.

Oracle GoldenGate Installed Components

This appendix describes the programs, directories, and other components created or used by the Oracle GoldenGate software in the Oracle GoldenGate installation directory. Additional files not listed here might be installed on certain platforms. Files listed here might not be installed on every platform.

This appendix includes the following sections:

- [Section A.1, "Oracle GoldenGate Programs And Utilities"](#)
- [Section A.2, "Oracle GoldenGate Subdirectories"](#)
- [Section A.3, "Other Oracle GoldenGate Files"](#)
- [Section A.4, "Oracle GoldenGate Checkpoint Table"](#)

A.1 Oracle GoldenGate Programs And Utilities

This section describes programs installed in the root Oracle GoldenGate installation directory.

Note: Some programs may not exist in all installations. For example, if only capture or delivery is supported by Oracle GoldenGate for your platform, the extract or replicat program will not be installed, respectively. Likewise, special files might be installed to support a specific database.

Table A-1 Oracle GoldenGate Installed Programs and Utilities

Program	Description
convchk	Converts checkpoint files to a newer version.
checklist.sql	Dumps your environment.
defgen	Generates data definitions and is referenced by Oracle GoldenGate processes when source and target tables have dissimilar definitions.
emscnt	Sends event messages created by Collector and Replicat on Windows or UNIX systems to EMS on NonStop systems.
extract	Performs capture from database tables or transaction logs or receives transaction data from a vendor access module.
ggminstall	Oracle GoldenGate installation script for the SQL/MX database.
ggsci	User interface to Oracle GoldenGate for issuing commands and managing parameter files.

Table A-1 (Cont.) Oracle GoldenGate Installed Programs and Utilities

Program	Description
ggsmgr.jcl ggsmgr.proc ggsmgrst.jcl ggsmgrst.proc	Start the Oracle GoldenGate Manager process from a batch job or the operator console on a z/OS system. Installed to support DB2 z/OS databases.
install	Installs Oracle GoldenGate as a Windows service and provides other Windows-based service options.
keygen	Generates data-encryption keys.
logdump	A utility for viewing and saving information stored in extract trails or files.
mgr	(Manager) Control process for resource management, control and monitoring of Oracle GoldenGate processes, reporting, and routing of requests through the GGSCI interface.
replicat	Applies data to target database tables.
reverse	A utility that reverses the order of transactional operations, so that Replicat can be used to back out changes from target tables, restoring them to a previous state.
server	The Collector process, an Extract TCP/IP server collector that writes data to remote trails.
triggen	Generates scripts that create the Oracle GoldenGate log table and logging triggers to support the trigger-based extraction method.
vamserv	Started by Extract to read the TMF audit trails generated by TMF-enabled applications. Installed to support the NonStop SQL/MX database.

A.2 Oracle GoldenGate Subdirectories

This Section describes the subdirectories of the Oracle GoldenGate installation directory and their contents.

Note: Some directories may not exist in all installations.

Table A-2 Oracle GoldenGate installed subdirectories

Directory	Description
br	Contains the checkpoint files for the bounded recover feature.
cfg	Contains the property and XML files that are used to configure Oracle GoldenGate Monitor.
dirdb	Contains the data store that is used to persist information that is gathered from an Oracle GoldenGate instance for use by the Oracle GoldenGate Monitor application or within Oracle Enterprise Manager.

Table A–2 (Cont.) Oracle GoldenGate installed subdirectories

Directory	Description
dirchk	<p>Contains the checkpoint files created by Extract and Replicat processes, which store current read and write positions to support data accuracy and fault tolerance. Written in internal Oracle GoldenGate format.</p> <p>File name format is <i>group_name+sequence_number.ext</i> where <i>sequence_number</i> is a sequential number appended to aged files and <i>ext</i> is either <i>cpe</i> for Extract checkpoint files or <i>cpr</i> for Replicat checkpoint files.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>ext1.cpe</p> <p>rep1.cpr</p>
dirdat	<p>The default location for Oracle GoldenGate trail files and extract files that are created by Extract processes to store extracted data for further processing by the Replicat process or another application or utility. Written in internal Oracle GoldenGate format.</p> <p>File name format is a user-defined two-character prefix followed by either a six-digit sequence number (trail files) or the user-defined name of the associated Extract process group (extract files).</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>rt000001</p> <p>finance</p>
dirdef	<p>The default location for data definitions files created by the DEFGEN utility to contain source or target data definitions used in a heterogeneous synchronization environment. Written in external ASCII. File name format is a user-defined name specified in the DEFGEN parameter file.</p> <p>These files may be edited to add definitions for newly created tables. If you are unsure of how to edit a definitions file, contact Oracle GoldenGate technical support.</p> <p>Example:</p> <p>defs.dat</p>
dirjar	<p>Contains the Java executable files that support Oracle GoldenGate Monitor.</p>
dirout	<p>This directory is not used any more.</p>
dirpcs	<p>Default location for status files. File name format is <i>group.extension</i> where <i>group</i> is the name of the group and <i>extension</i> is either <i>pce</i> (Extract), <i>pcr</i> (Replicat), or <i>pcm</i> (Manager).</p> <p>These files are only created while a process is running. The file shows the program name, the process name, the port number, and the process ID.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>mgr.pcm</p> <p>ext.pce</p>

Table A–2 (Cont.) Oracle GoldenGate installed subdirectories

Directory	Description
dirprm	<p>The default location for Oracle GoldenGate parameter files created by Oracle GoldenGate users to store run-time parameters for Oracle GoldenGate process groups or utilities. Written in external ASCII format. File name format is <i>group name/user-defined name.prm</i> or <i>mgr.prm</i>.</p> <p>These files may be edited to change Oracle GoldenGate parameter values after stopping the process. They can be edited directly from a text editor or by using the <code>EDIT PARAMS</code> command in GGSCI.</p> <p>Examples:</p> <p><code>defgen.prm</code></p> <p><code>finance.prm</code></p>
dirrec	Not used by Oracle GoldenGate.
dirrpt	<p>The default location for process report files created by Extract, Replicat, and Manager processes to report statistical information relating to a processing run. Written in external ASCII format.</p> <p>File name format is <i>group name+sequence number.rpt</i> where <i>sequence number</i> is a sequential number appended to aged files.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p><code>fin2.rpt</code></p> <p><code>mgr4.rpt</code></p>
dirsql	Used by the <code>triggen</code> utility to store SQL scripts before <code>triggen</code> was deprecated. Currently used to store training scripts and any user-created SQL scripts that support Oracle GoldenGate.
dirtmp	The default location for storing transaction data when the size exceeds the memory size that is allocated for the cache manager. Do not edit these files.
dirwlt	Contains the Oracle Wallet that supports Oracle GoldenGate Monitor. This directory is not installed until you run the utility that creates the wallet.
UserExitExamples	Contains sample files to help with the creation of user exits.

A.3 Other Oracle GoldenGate Files

This section describes other files, templates, and objects created or installed in the root Oracle GoldenGate installation directory.

Note: Some files may not be installed in your environment, depending on the database and OS platform.

Table A–3 Other Oracle GoldenGate Installed Files

Component	Description
<code>bcpfmt.tpl</code>	Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.
<code>bcrypt.txt</code>	Blowfish encryption software license agreement.

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
cagent.dll	Contains the Windows dynamic link library for the Oracle GoldenGate Monitor C sub-agent.
category.dll	Windows dynamic link library used by the install utility.
chkpt_db_create.sql	Script that creates a checkpoint table in the local database. A different script is installed for each database type.
db2cntl.tpl	Template for use with Replicat when creating a control file for the IBM LOADUTIL bulk-load utility.
ddl_cleartrace.sql	Script that removes the DDL trace file. (Oracle installations)
ddl_ddl2file.sql	Script that saves DDL from the marker table to a file.
ddl_disable.sql	Script that disables the Oracle GoldenGate DDL trigger. (Oracle installations)
ddl_enable.sql	Script that enables the Oracle GoldenGate DDL trigger. (Oracle installations)
ddl_filter.sql	Script that supports filtering of DDL by Oracle GoldenGate. This script runs programmatically; do not run it manually.
ddl_nopurgeRecyclebin.sql	Empty script file for use by Oracle GoldenGate support staff.
ddl_ora9.sql ddl_ora10.sql ddl_ora11.sql ddl_ora10upCommon.sql	Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.
ddl_pin.sql	Script that pins DDL tracing, the DDL package, and the DDL trigger for performance improvements. (Oracle installations)
ddl_purgeRecyclebin.sql	Script that purges the Oracle recyclebin in support of the DDL replication feature.
ddl_remove.sql	Script that removes the DDL extraction trigger and package. (Oracle installations)
ddl_session.sql ddl_session1.sql	Supports the installation of the Oracle DDL objects. This script runs programmatically; do not run it manually.
ddl_setup.sql	Script that installs the Oracle GoldenGate DDL extraction and replication objects. (Oracle installations)
ddl_status.sql	Script that verifies whether or not each object created by the Oracle GoldenGate DDL support feature exists and is functioning properly. (Oracle installations)
ddl_staymetadata_off.sql ddl_staymetadata_on.sql	Scripts that control whether the Oracle DDL trigger collects metadata. This script runs programmatically; do not run it manually.
ddl_trace_off.sql ddl_trace_on.sql	Scripts that control whether DDL tracing is on or off.
ddl_tracelevel.sql	Script that sets the level of tracing for the DDL support feature. (Oracle installations)
debug files	Debug text files that may be present if tracing was turned on.
demo_db_scriptname.sql demo_more_db_scriptname.sql	Scripts that create and populate demonstration tables for use with tutorials and basic testing.

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
.dmp files	Dump files created by Oracle GoldenGate processes for tracing purposes.
ENCKEYS	User-created file that stores encryption keys. Written in external ASCII format.
exitdemo.c	User exit example.
exitdemo_utf16.c	User exit example that demonstrates how to use UTF16 encoded data in the callback structures for information exchanged between the user exit and the process.
freeBSD.txt	License agreement for FreeBSD.
ggmessage.dat	Data file that contains error, informational, and warning messages that are returned by the Oracle GoldenGate processes. The version of this file is checked upon process startup and must be identical to that of the process in order for the process to operate.
ggserr.log	File that logs processing events, messages, errors, and warnings generated by Oracle GoldenGate.
ggsmg.dll	Windows dynamic link library used by the install program.
GLOBALS	User-created file that stores parameters applying to the Oracle GoldenGate instance as a whole.
help.txt	Help file for the GGSCI command interface.
icudt38.dll icuin38.dll icuuc38.dll	Windows shared libraries for International Components for Unicode.
jagent.bat	Windows batch file for the Java Agent for Oracle GoldenGate Monitor.
jagent.log jagentjni.log	Log files for the Oracle GoldenGate Monitor Agent.
jagent.sh	UNIX shell script for the Java Agent for Oracle GoldenGate Monitor
LGPL.txt	Lesser General Public License statement. Applies to free libraries from the Free Software Foundation.
libodbc.so	ODBC file for Ingres 2.6 on Unix.
libodbc.txt	License agreement for libodbc.so.
libxml2.dll	Windows dynamic link library containing the XML library for the Oracle GoldenGate XML procedures.
libxml2.txt	License agreement for libxml2.dll.
marker.hist	File created by Replicat if markers were passed from a NonStop source system.
marker_remove.sql	Script that removes the DDL marker table. (Oracle installations)
marker_setup.sql	Script that installs the Oracle GoldenGate DDL marker table. (Oracle installations)
marker_status.sql	Script that confirms successful installation of the DDL marker table. (Oracle installations)
notices.txt	Third-party software license file.

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
odbcinst.ini	Ingres 2.6 on Unix ODBC configuration file.
params.sql	Script that contains configurable parameters for DDL support. (Oracle installations)
pthread-win32.txt	License agreement for pthread-VC.dll .
pthread-VC.dll	POSIX threads library for Microsoft Windows.
prvtclkm.plb	Supports the replication of Oracle encrypted data.
pw_agent_util.bat	Script files that support the Oracle GoldenGate Monitor Agent.
pw_agent_util.sh	
role_setup.sql	Script that creates the database role necessary for Oracle GoldenGate DDL support. (Oracle installations)
sampleodbc.ini	Sample ODBC file for Ingres 2.6 on UNIX.
sqlldr.tpl	Template for use with Replicat when creating a control file for the Oracle SQL*Loader bulk-load utility.
start.prm	z/OS paramlib members to start and stop the Manager process.
stop.prm	
startmgr	z/OS Unix System Services scripts to start the Manager process from GGSCI.
stopmgr	
startmgrcom	z/OS system input command for the Manager process.
stopmgrcom	
tcperrs	File containing user-defined instructions for responding to TCP/IP errors.
usrdecs.h	Include file for user exit API.
xerces-c_2_8.dll	Apache XML parser library.
zlib.txt	License agreement for zlib compression library.

A.4 Oracle GoldenGate Checkpoint Table

When database checkpoints are being used, Oracle GoldenGate creates a checkpoint table with a user-defined name in the database upon execution of the ADD CHECKPOINTTABLE command, or a user can create the table by using the `chkpt_db_create.sql` script (where `db` is an abbreviation of the type of database that the script supports).

Do not change the names or attributes of the columns in this table. You can change table storage attributes as needed.

Table A-4 Checkpoint Table Definition

Column	Description
GROUP_NAME (primary key)	The name of a Replicat group using this table for checkpoints. There can be multiple Replicat groups using the same table.
GROUP_KEY (primary key)	A unique identifier that, together with GROUPNAME, uniquely identifies a checkpoint regardless of how many Replicat groups are writing to the same table.
SEQNO	The sequence number of the checkpoint file.

Table A-4 (Cont.) Checkpoint Table Definition

Column	Description
RBA	The relative byte address of the checkpoint in the file.
AUDIT_TS	The timestamp of the checkpoint position in the checkpoint file.
CREATE_TS	The date and time when the checkpoint table was created.
LAST_UPDATE_TS	The date and time when the checkpoint table was last updated.
CURRENT_DIR	The current Oracle GoldenGate home directory or folder.